

EXTENDED BASIC for LASER 1101210/310

EXTENDED BASIC They brought a large number of new and more comfortable BASIC commands.

On the disk there are two different versions, which differ in the size of the memory required, but also in the number of commands.

The short version with a storage requirement of about 1200 bytes contains the following 30 commands:

AUTO, DELETE, TRON, TROFF, MEM, FRE, ERROR, ERR, ERL, DEFINT, DEFSGN, DEFDBL, DEFSTR, RANDOM, ON, RESUME, VARPTR, SYSTEM, STRING \$, POS, VLOAD, MERGE, COMPRESS, RENEW, CINT, CSNG, CDBL, FIX, CALL, memSize. In the large version with a storage requirement of approximately 3500 bytes, the following nine commands are included in addition: RENUM, PLOT, NPLOT, CIRCLE, RECT, PAINT. CPoint, GCL, LPEN. EXTENDED BASIC is upward-compatible with the original BASIC version of your computer and also with the BASIC-UP; ie programs created there can be taken in EXTENDED BASIC, edited and even started.

In EXTENDED BASIC acquired programs are no longer run under the normal BASIC or BASIC-UP.

Between the two versions of the EXTENDED BASIC programs can be exchanged. They are only executable in the short version if none of the nine special commands the large version were used. Loading EXTENDED BASIC with the BRUN command.

It logs after a successful download a short Vorstelltext and the READY message. You can then start using the normal BASIC processing.

Entering the new BASIC commands are normal, ie without any identification.

Annotation:

In the below-described syntax of the commands a new string enclosed in square brackets ([xxx]) represents an optional entry.

Description of the new commands

1. To support the program execution YLOAD [" name "] With this command you can read a BASIC program from tape, that was not created under EXTENDED BASIC. The program is automatically adjusted to the address range of the respective version After downloading, you can start this program, listen, edit and, if necessary. Save on tape. CSAVE with the EXTENDED BASIC composure. So stored programs can then be loaded as usual or CLOAD or CRUN restarted under EXTENDED BASIC.

Danger:

All programs are not created with EXTENDED BASIC must be adapted with VLOAD first, even at the transition from one version of BASIC EXTENDED to the other. The direct loading these programs CLOAD or CRUN usually leads to destruction of the EXTENDED BASIC.

MERGE [" Surname"]

With this command a second BASIC program can be attached to an already placed in storage. This program does not need to be created using the same BASIC version. However, the line numbers of the second program should be greater than that in the memory, if necessary, renumber the previously with RENUM.

RENUM [n, s) (only in the large version) RENUM

[n, s]

This command is the renumbering of a BASIC program. With "n", the initial value, the step size are marked with "s" can. Missing one of the two parameters, as is employed as StandardwertJ0.

"Example: RENUM 100.5 renumbering starting from 100 in steps of 5 renumbering starting from 10 in 10 steps..

AUTOMOBILE [n, s]

This command automatically generated prior line numbers
Program line that you want to enter. With "n" can be an initial value and

everyone

with "s" indicating the increment. The default value is numbered in increments of 10 from the 10th

You leave the AUTO function by pressing the CTRL and BREAK keys.

Pushes the computer is in AUTO mode to an already occupied line, the line contents is displayed and can be overwritten or unchanged solely by actuating the RETURN key.

Danger:

Contains such already occupied cell one of the additional commands EXTENDED BASIC, so they can not be shown. D. h. they will be lost if you edit such a line or back ends with RETURN.

You are not sure, then you leave the AUTO mode, and check the line with LIST.

Example: AUTO 10.5 line numbering from 10 in steps of 5.

DELETE Line number [-Zeilennummer]

This command deletes the specified row or the row area. Example: DELETE 20

Line 20 deletes
DELETE deletes rows 50-100 50-100
DELETE deletes -40 from the beginning of the program up to the line 40

COMPRESS

This all spaces are removed (except for strings) and "Remarks" from the program text.

This can lead to a significant reduction in program length and a shorter charging time.

You should look at the developed programs you a neatly written and documented version For Troubleshooting and keep development on tape. To run, however, the compressed form of offers, because it is faster loaded, the memory optimally utilized and runs faster. **RENEW** Once you have deleted a program text from accidentally entering NEW, so you can restore it with the help of RENEW commands.

However, this only works if you have not already begun with a new input (RUN LIST or have no effect).

TRON

Switch on the trace function. It allows you to track the progress of a program for troubleshooting. Once the computer is running a new program line, its line number is displayed in brackets. TROFF off the TRACE function.

Annotation: TRON and TROFF can also be employed within a program become, if only the procedure one certain Program section is desired.

memSize address

With this command you can set the highest possible available for BASIC address. This gives you the opportunity where you want to put a machine program routine a storage area to protect from destruction by automatically BASIC memory management. As address the lowest accessible BASIC memory address and a maximum of the top address of your memory configuration is minimal specify. The address specification is decimal.

This command can be used within a BASIC program. It is important to make sure that it is done as one of the first commands that all variables are cleared (implicitly contains the command CLEAR

50). Example: MemSize 42000

SYSTEM

This command your computer is re-initialized. After this command neither the EXTENDED BASIC still possibly entered program are available.

This is useful if you want to continue with the normal BASIC subsequently or want to load a machine program.

11. Within a Program

DEFINT Letter range variables with a letter from the defi

start-defined range of letters, are whole numbers (integers) and stored.

So defined variable show in memory only 2 bytes. Note that an integer variable can only have values from -32768 to +32767.

10 DEFINT N, R, V or 10 DEFINT AF DEFSGN letter range

Variables that begin with a letter from the letter range will be treated as a variable single-precision and stored. Variables and constants single precision are stored with 7 points and issued with six points.

Note that all numeric variables, unless otherwise defined, single precision have (standard). DEFSGN is useful if different from previously defined variable to be redefined within a program. 10 DEFSGN AE, X

DEFDBL letter range

Variables that begin with a letter from the letter range are treated as variable double-precision and stored. One such variable is handled internally by 17 points, which are outputted 16

10 DEFDBL G

DEFSTR letter range

Variables that begin with a letter, this letter range are treated as a string variable and saved.

annotation

An explicit definition in the program for. B. having% for integer or \$ for String, overrides a decision taken definition. 10 DEFDBL AE 20 A 10%

30 A \$ "ABCD" 40 AX 100D4

A% ~ Integer, A \$ = string, AX = double precision

ERROR

This command is used to branch into an ERROR routine.

Application: 10 ON ERROR
GOTO 100

In case of failure the program branches to line 100 where an appropriate error evaluation and handling can take place. ERROR will appear in the command position (at the beginning of a line or after a ;), interpreted this as your computer SOUND command.

RESUME (Param)

This command terminates an error handling routine and specifies where normal program execution should be continued. RESUME 0 or RESUME effected without line number that the program is continued in the line in which the error occurred. RESUME with line number

(Z. B. RESUME 50) causes a Program continuation in this line. RESUME NEXT program causes a continuation of the line that is to the faulty line. ERL Returns the line number where the error occurred. ERR Returns error code of the error back. Example Program for the application of error handling: 10 10 20 CLEAR ON ERROR GOTO 1000 30 INPUT "GIVE MESSAGE ON"; M \$ 40 INPUT NOW a digit PLEASE "N 50 Z = YN 60 PRINT" ENTER VALUES CORRECT

TRY SOMETHING

WRONG. "70

GOTO 30

1000 IF ERL = 30 AND ERR = 26 THEN 1040 1010 IF ERL = 40 AND ERR = 10 THEN 1050 1020 IF ERL = 50 AND ERR = 20 THEN GOTO 1060 1030 ERROR ON 0: RESUME

1040 PRINT "MESSAGE Is_Closed LONG - MAX 10ZEICHEN." RESUME 1050 PRINT "DIGIT TOO LARGE": RESUME 1060 PRINTJEILEN BY 0 IN LINE 50 - POINT DIFFERENCE 0 fillin '1070 RESUME 40

Error numbers and corr. Messages 00 = NEXT
WITHOUT FOR 02 = SYNTAX ERROR

04 = RETURN WITHOUT GOSUB 06 =
OUT OF DATA
08 = 10 = Illegal Function Call
OVERFLOW = 12 OUT OF MEMORY 14
= UNDEFINED LINE

16 = SUBSCRIPT OUT OF RANGE 18 = 20 =
ARRAY REDIMENSIONED DIVISION BY
ZERO
22 = ILLEGAL DIRECT OPERATION 24 =
TYPE MISMATCH = 26 OUT OF SPACE 28
STRING = STRING TOO LONG

30 = 32 = STRING Formula too complex CAN NOT
CONTINUE 34 = NO RESUME

36 = 38 = RESUME WITHOUT ERROR
ERROR unprintable 40 = MISSING
OPERAND 42 = BAD DATA FILE

44 = DISK BASIC COMMAND

MEM Reports the size of the available free space. Example: 200 IF MEM < 200 THEN 700 If MEM
is to be used as a command, it must be run in conjunction with PRINT. PRINT MEM returns the
number of bytes in memory in which no program variables or strings are.

FRE [Param] FRE (0) is synonymous with MEM (see above) FRE ("") returns for character
variables (strings) is still available space. This value is calculated from the reserved area with
CLEAR less is already stored character variables. Examples: CLEAR 100: PRINT FRE output:
100 CLEAR 100: A \$ = "ABCDE" PRINT FRE ("") Issue: 95

VARPTR (Variable) The address to which the value of the specified variable is passed. Example:
10 K = VARPTR (A) for different types of variables has "K" following meaning: a) integer variable
(integer) (A%) K = LSB K + 1 = MSB b) Variable single precision (A) K = LSB K + 1 = next.
Wertbyte K + 2 = MSB + K 3 = exponent c) Variable double precision (DEFDBL A) K = K LSB + 1 =
next. Wertbyte K + 6 MSB K + 7 exponent d) string variables (A \$) K = length of the string K + 1
LSB of the starting address K + 2 MSB of the start address LSB is the least significant byte MSB is
the highest-significant byte ON n GOTO Zeile1, line2 ... lines ON n GOSUB Zeile1, line2 rows This
command allows to simultaneously specify multiple branch destinations. is bound according to the
content of the variable "n". The value of "n" has to be between 0 to 255. If an ON - ON GOTO or -
GOSUB command is executed, the integer portion of "n" is first determined. Now, the computer
scans the nth element in the line number list and jump to that line number. following GOSUB
command is executed - "n" is equal to 0 or greater than the number of specified line numbers, the
ON GOTO ON or is. When "n" is less than 0, an error occurs. In an ON - GOSUB is made to return

the subroutine (RETURN) the program continues to the ON-GOSUB following command. Example:
Without ON

50 IF C = 1 GOTO 200 60 IF C = 2 GOTO 300 70 IF C = 3 GOTO 400 80 IF C = 4 GOTO 500 90 IF
C <1 OR C> 4 GOTO 1 00: REM next command with ON

50 ON C GOTO 200,300,400,500 100 ...

The use of ON in conjunction with ERROR ERROR see description of command. POS (0)

It is determined the current position of the cursor (pointer) in the line and returned. 10 A = POS (0)

The argument (0) must be specified, but is not significant.

RANDOM

When executing this command, a new starting value for the determination of random numbers RND (x) is set.

STRING \$ in, character or number)

Creates a string consisting of n times the specified characters. Example:

10 A \$ = STRING \$ (10, "-") after

executing = A \$ ".....

and the corresponding ASCII code can be entered directly as a number instead of the character.

10 A \$ = STRING \$ (8,35) according to embodiment A \$ = "

CINT (X) Calculates the nearest whole number which is less than the argument X. The argument must be between -32768 and +32767. Example: A = CINT (2.6) gives A 2 A = CINT (-2.6) yields A

3 CSNG (X) Creates simple accurate representation of X. Calculates is a 6-digit number with 4/5 rounding in double-precision X 9

CDBL (X)

Generates the double-precision representation of X. The result gets 17 points, of which the sites containing the argument X, are significant. FIX (X)

Separates the decimals on X. A = FIX (1.5) returns

1 A = FIX (-1.5) gives A

CALL address

Jump to the given address, this command

is a convenient way to call from Machine program routines from a BASIC program. Unlike USR, the start address

directly decimal in command specify.

The machine program must be terminated with RET, a Parameter transfer is not possible.

Example: CALL 457 clears the screen (CLS) III. Additional graphics commands The commands listed below are included only in the large version of the BASIC EXTENDED.

They will only run when the computer is in graphics mode. (MODE (1 "Otherwise, an error message is issued.

GCL [n]

Clear the screen memory for high-resolution graphics with the specified as "n" color. "N" can assume the values of 1-4.

"N" is not specified, the image memory is cleared with the background color. Example:

MODE 10 (1): 3 GCL paints the screen in blue. cPoint (x, y) extending the POINT

command.

Here you do not get not only the information, whether a particular point in the storage of high-resolution graphics or law, but you will also be the relevant color code (1-4) informed (1 = background). Example:

MODE 10 (1): 3.0 COLOR SET 20 (20,30) 30 A =

CPoint (20,30)

After execution of this instruction sequence A contains the value of 3.

PLOT [Cn] Plotting information sets a point or draws a line on the screen. PLOT information can take several formats. a) PLOT X, Y is the point with the coordinates X and Y (corresp. to the SET command). b) PlotX, Y TO X1, Y1 draws a line from X, Y to X1, Y1 c) PLOT X, Y TO X1, Y1 TO X2, Y2 TO Xn, Yn draws a line of XX to X1, Y1 from there to X2X2, etc. to Xn, Yn. d) PLOT to X1, Y1 TO draws a line from the target point of the last PLOT command to X1X1, etc. for "Cn" may optionally contain a color to be specified. This information overwrites the agreement with COLOR color choice (n = 1 -4) for this command. Is "cn" is not specified, the color of the last COLOR command is used. Example: 50 PLOT C2,

20.30 TO 70.50 pulling a blue line from 20.30 to 70.50

NPLOT

Corresponds to the PLOT command, except that the line is reset.

CIRCLE [Cn] X, Y, R draws a circle whose center is the coordinates X and Y and its radius R is. Exceeds einerder3 values 255 is an error message. Cn a color can be selected as the PLOT command that overrides the agreement with COLOR choice.

RECT [Cn] X1, Y1 TO X2, Y2 Draws a rectangle on the screen, wherein X1 and Y1 specify the Koordianten the upper left corner and X2X2 the coordinates of the lower right corner. Additional color choices Cn as with PLOT and CIRCLE.

PAINT X, Y, BI [, B2, B3] This command can color areas on the screen. The starting point is the coordinates X, Y. With BI, B2 11

and B3 up to three boundary colors can be specified. BI is then simultaneously used as fill color. Example:

It is to be drawn on a red background, a yellow circle imagine blue. The center of the circle has the coordinates X = 64 and Y = 32, the radius should be 15 °.

MODE 10 (1): COLOR 3.0: 'switch to Figure 20 GCL 4:' delete screen with red

Draw 'circle: CIRCLE 30 C2, 64, 32, 15

PAINT 40 64, 32, 3, 2 'blue coloring circuit 50 GOTO 50' endless loop

IV. Use of a light pen (light pen) This command also exists only in the major version (BRUN "LPEN" or RUN "DEMO")

LPEN (X, Y [, A])

This command determined coordinates and, optionally, the image address of the screen on which a light pen is placed.

numeric variable to be specified for X and Y, in which the XY coordinates are transmitted from the LPEN command.

Parameter "A" is optional. here is also a numeric variable is defined, the screen address is passed in this addition. If the light pen is not

properly set up, or tune the Brightness ratios of the screen, the X is not an evaluation, they will set Y variables to 255 and A on -'I.

The values for X, Y and A are according to the respective mode is determined (graphics or text) that is

MODE (0) X = Y = 0-31 0-15A = 0-511

MODE (1) X = Y = 0-127 0-63 A = 0-2047 example:

In Text mode should at each contact point of the light pen, the character "-" appear on the screen 10 CLS.

20 LPEN (X, Y, A) 30 IF X = 255 THEN 20

40 SOUND 33, 1: PRINT @ A, "" '50 GOTO 20